Instituto Tecnológico de Chihuahua ll

Clase: Programación Web Front End

Carrera: Ingeniería en Sistemas Computacionales

Maestro: Carlos Humberto Rubio

Alumnos y Números de Control:

Nicole Johana Bibiano Ochoa #19550731

Emmanuel Chavira Holguín #19550730

Luis Antonio Amador Estrada #19550793

Mario Alonso Meza Alvidrez #19550812

Fecha: 27/Febrero/2022

Material: Documento de Presentación de Bazaar

Índice

[Introducción 1](#_Toc96894542)

[Objetivo 1](#_Toc96894543)

[Desarrollo 1](#_Toc96894544)

[Conceptos Generales de un SCV 3](#_Toc96894545)

[Diferentes Plataformas 5](#_Toc96894546)

[Versiones 5](#_Toc96894547)

[Pasos para instalar 6](#_Toc96894548)

[Plugins 8](#_Toc96894549)

[Comandos 9](#_Toc96894550)

[De quien sucede Bazaar 10](#_Toc96894551)

[Algunas características especiales 10](#_Toc96894552)

[Conclusiones 10](#_Toc96894553)

[Referencias: 11](#_Toc96894554)

# Introducción

A continuación voy a redactar el documento de la exposición que hicimos mis compañeros de equipo y yo el día Jueves 24 de Febrero del 2022 donde pondremos toda la información que nos faltó o no pudimos poner en la presentación por cuestión de tiempo

# Objetivo

El objetivo de este documento es que los alumnos conozcan más a fondo el sistema de control de versiones Bazaar, como se usa y que beneficios nos puede ofrecer a nosotros como programadores al usarlo o saber utilizarlo

# Desarrollo

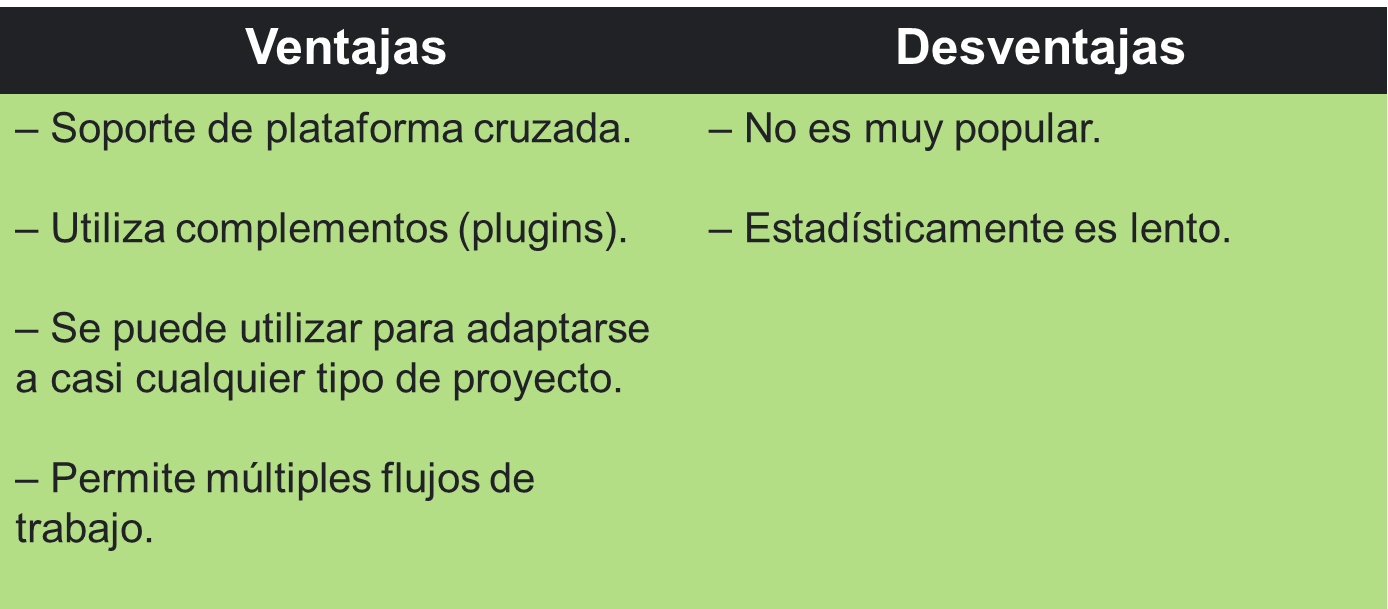
Bazaar es un sistema de control de versiones distribuido, diseñado para facilitar la contribución en proyectos de software libre y opensource.

Puede ser usado por un usuario único trabajando en múltiples ramas de un contenido local, o por un equipo colaborando a través de la red.

Bazaar está escrito en lenguaje de programación Python y tiene versiones empaquetadas para la mayoría de distribuciones GNU/Linux, así como Mac OS X y MS Windows. Bazaar es software libre y parte del proyecto GNU.

El creador de Bazaar es Martin Pool y fue lanzado el 14 de diciembre de 2007 con la versión 1.0.

Bazaar está escrito en lenguaje de programación Python 2 y Pyrex.



Las ventajas principales de utilizar un SCV, son:

Espacio de almacenamiento centralizado de, principalmente, el código fuente de la aplicación así como scripts de construcción -en el caso de aplicaciones que requieran ser compiladas o simplemente, necesiten realizar configuraciones especiales, ya sea tanto para continuar desarrollándolas como para ejecutarlas-.

Para ser efectivos, deben llevar un control histórico de cambios que se vayan efectuando en los archivos -preferentemente automático-, permitir el establecimiento de tags -etiquetas- que ayuden a identificar diferentes releases -versiones-.

Los Sistemas de Control de Versiones (SCV) pueden agruparse en dos tipos:

**Centralizados:** Un único repositorio centralizado administrado por un solo responsable.

**Distribuidos (recomendados):** Donde existe un repositorio central que cada usuario podrá clonar para obtener su propio repositorio -local- e interactuar con otros repositorios locales.

# Conceptos Generales de un SCV

**Repositorio**

Es la copia maestra donde se guardan todas las versiones de los archivos de un proyecto.

**Copia de trabajo**

La copia de los ficheros del proyecto que podemos modificar. Cada desarrollador tiene la suya propia.

**Check Out / Clone**

La acción empleada para obtener una copia de trabajo desde el repositorio. En los scv distribuídos -como Git- esta

operación se conoce como clonar el repositorio por que, además de la copia de trabajo, proporciona a cada programador

su copia local del repositorio a partir de la copia maestra del mismo.

**Check In / Commit**

La acción empleada para llevar los cambios hechos en la copia de trabajo a la copia local del repositorio -CheckIn-. Esto

crea una nueva revisión de los archivos modificados. Cada commit debe ir acompañado de un Log Message el cual es un

comentario -Una cadena de texto que explica el commit- que añadimos a una revisión cuando hacemos el commit oportuno.

**Push**

La accion que traslada los contenidos de la copia local del repositorio de un programador a la copia maestra del mismo.

**Update/Pull/Fetch+Merge/Rebase**

Acción empleada para actualizar nuestra copia local del repositorio a partir de la copia maestra del mismo, además de

actualizar la copia de trabajo con el contenido actual del repositorio local.

**Conflicto**

Situación que surge cuando dos desarrolladores hacen un commit con cambios en la misma región del mismo fichero. El

scv lo detecta, pero es el programador el que debe corregirlo.

**Etiquetar**

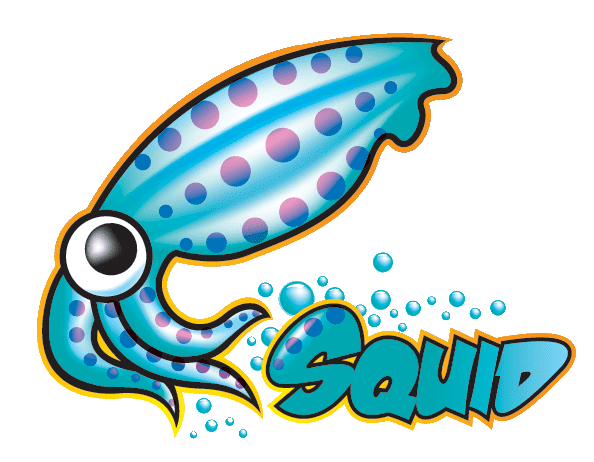
Poner un nombre común a todos los ficheros bajo control de versiones de un proyecto en un instante determinado

**Rama**

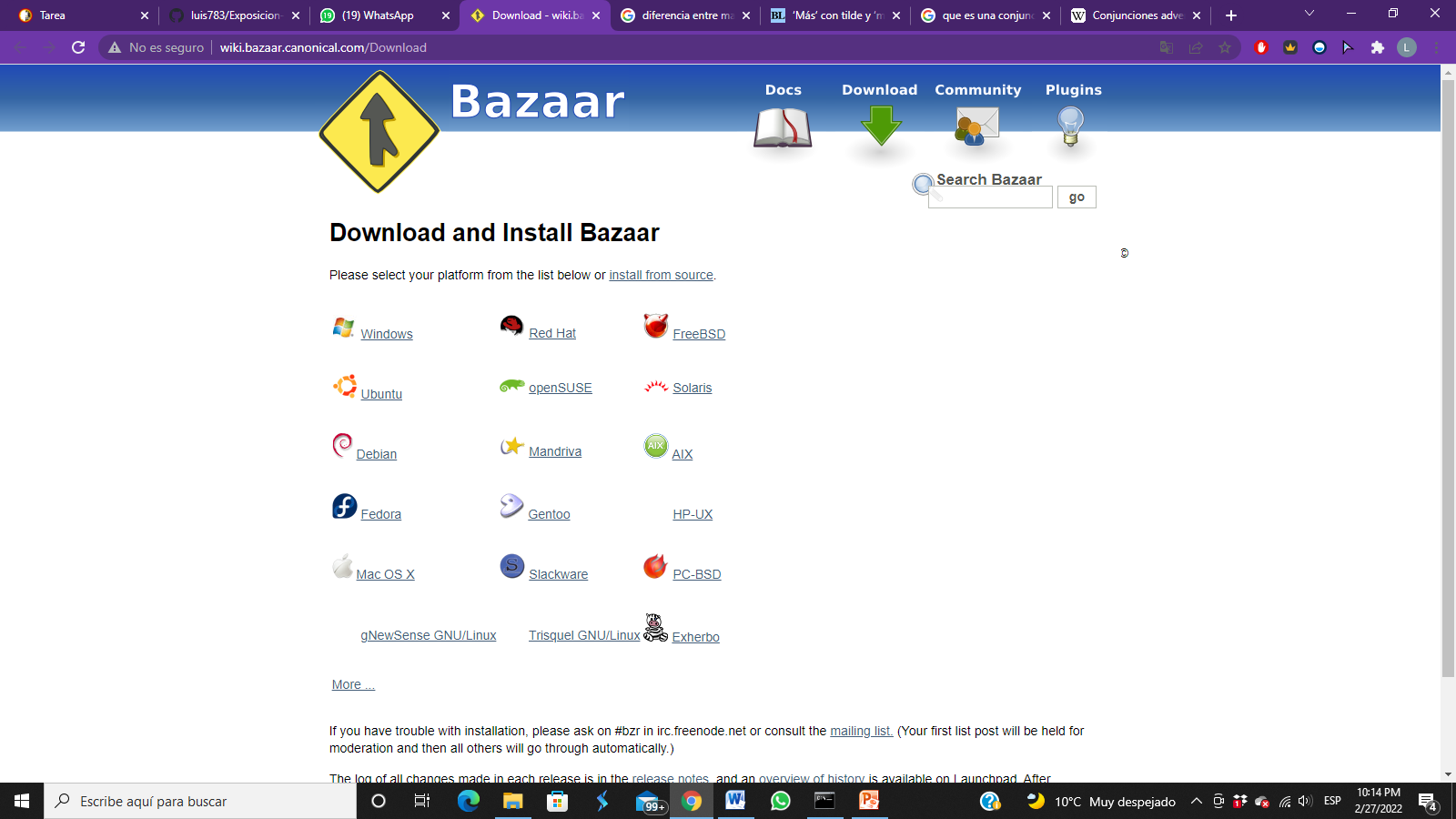
Representa una línea de evolución de nuestro software. Podemos tener todas las que queramos, pero normalmente una es

especial: trunk, master/main, etc. . .

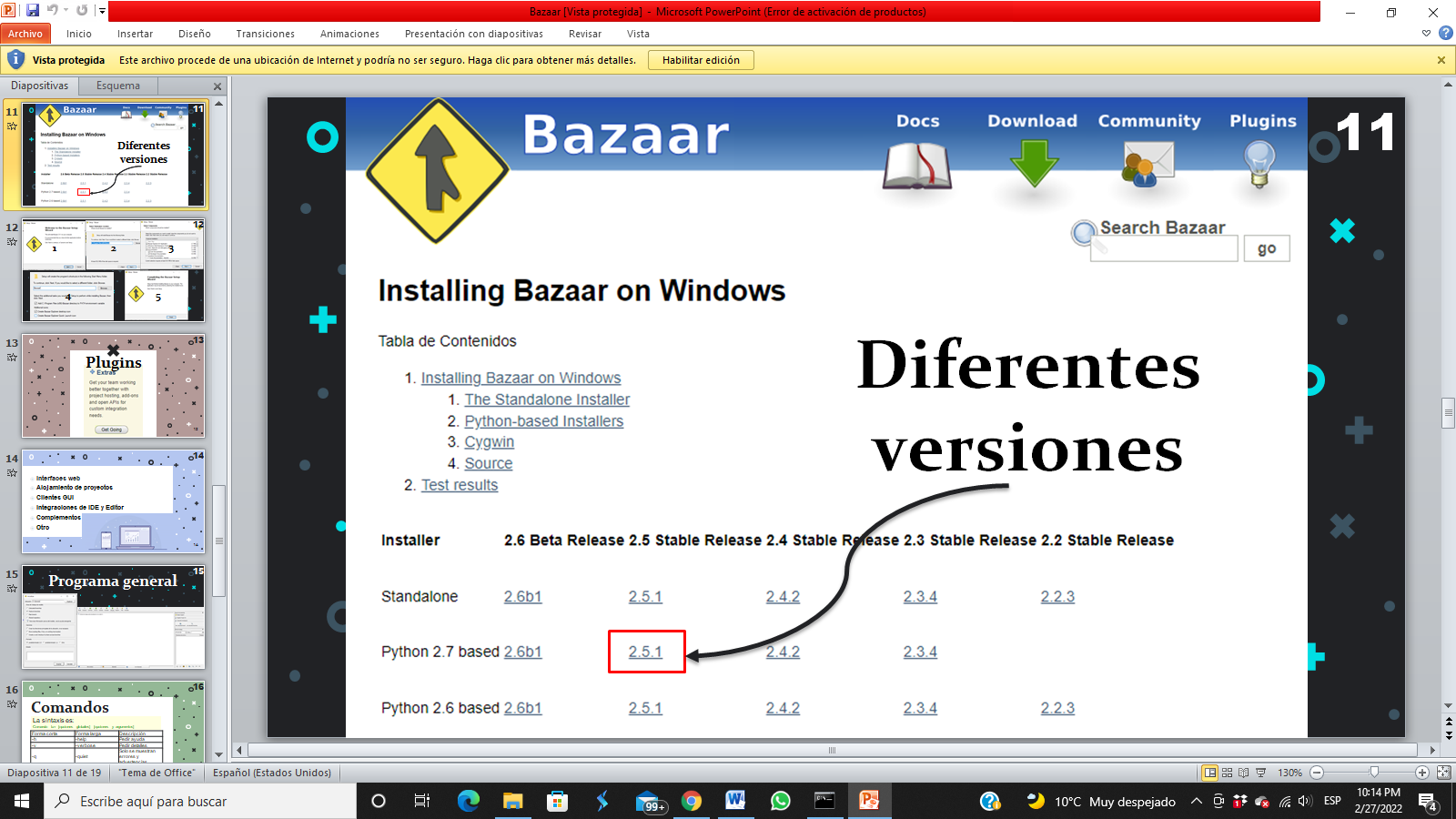
Proyectos destacados que utilizan Bazaar como sistema de control de versiones:



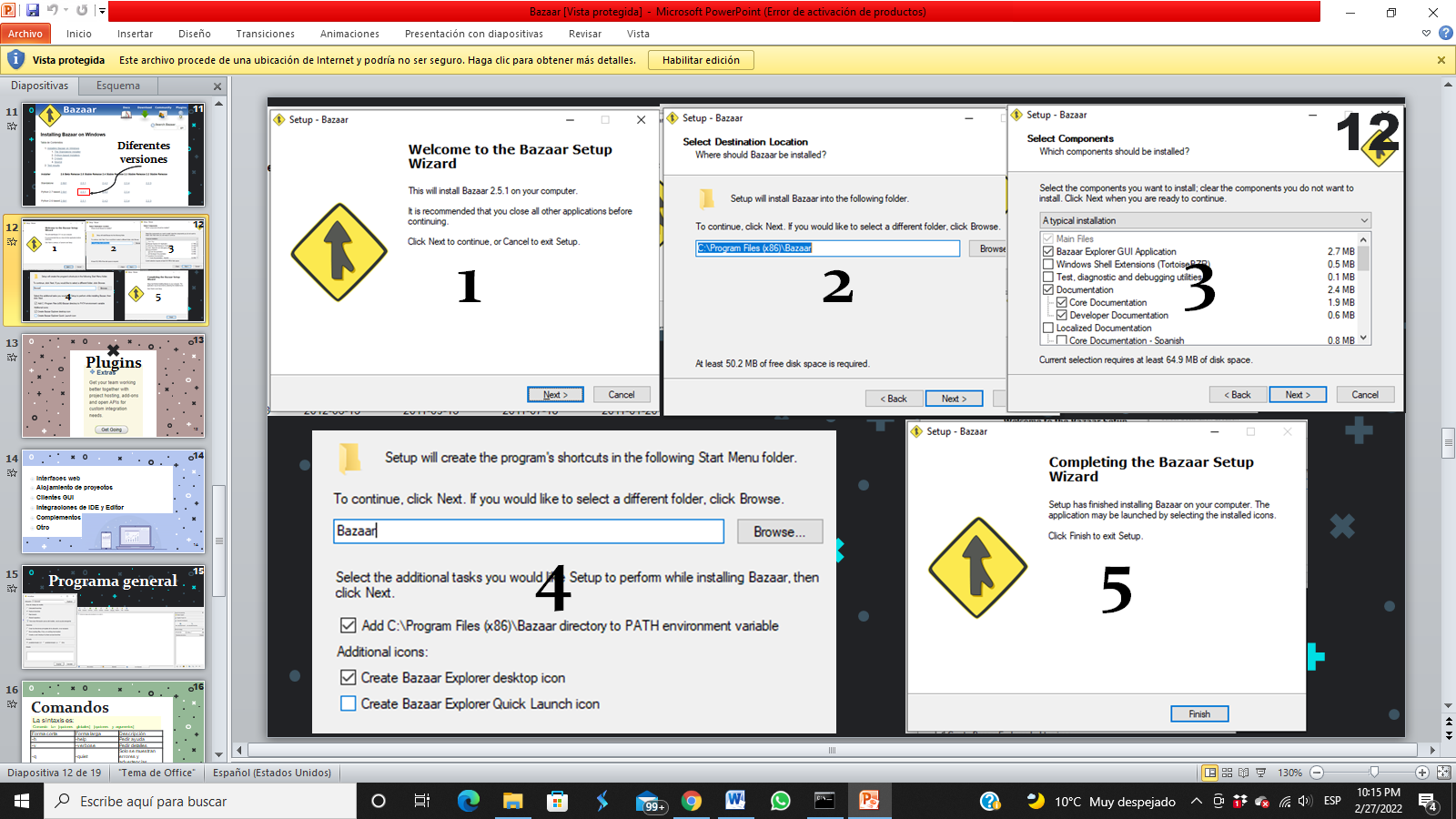
# Diferentes Plataformas



# Versiones



# Pasos para instalar



Instalando Bazaar.

Para instalar Bazaar en tu computadora, ejecuta el siguiente comando:

sudo apt-get install bzr

1. Una vez instalado Bazaar, deberás clonar el repositorio central a tu computadora local:

bzr branch sftp://tu\_usuario@IP\_Servidor\_SCV /home/tu\_usuario/public/trunk

2. A continuación, deberás ingresar tu contraseña.

3. Una vez clonado el repositorio, agregas unas líneas al archivo de configuración de Bazaar. Para ello, abre el archivo de configuración con el editor Nano:

nano trunk/.bzr/branch/branch.conf

4. Mueve el cursor hasta la siguiente línea y pulsa las teclas Ctrl + K:

parent\_location = sftp://tu\_usuario@IP\_Servidor\_SC /home/tu\_usuario/public/trunk

5. A continuación, pulsa tres veces, las teclas Ctrl + U para pegar (tres veces), la línea que cortaste anteriormente. Deberás ver lo siguiente:

parent\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

parent\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

parent\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

6. Reemplaza la palabra “parent” de la segunda línea, por “push” y la de la tercera, por “pull” de forma tal que el archivo, se vea como sigue:

parent\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

push\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

pull\_location = sftp://tu\_usuario@ IP\_Servidor\_SC /home/tu\_usuario/public/trunk

7. Para guardar el archivo pulsa las teclas Ctrl + O (enter) y para salir, pulsa Ctrl + X.

Listo!

# Plugins

Interfaces web

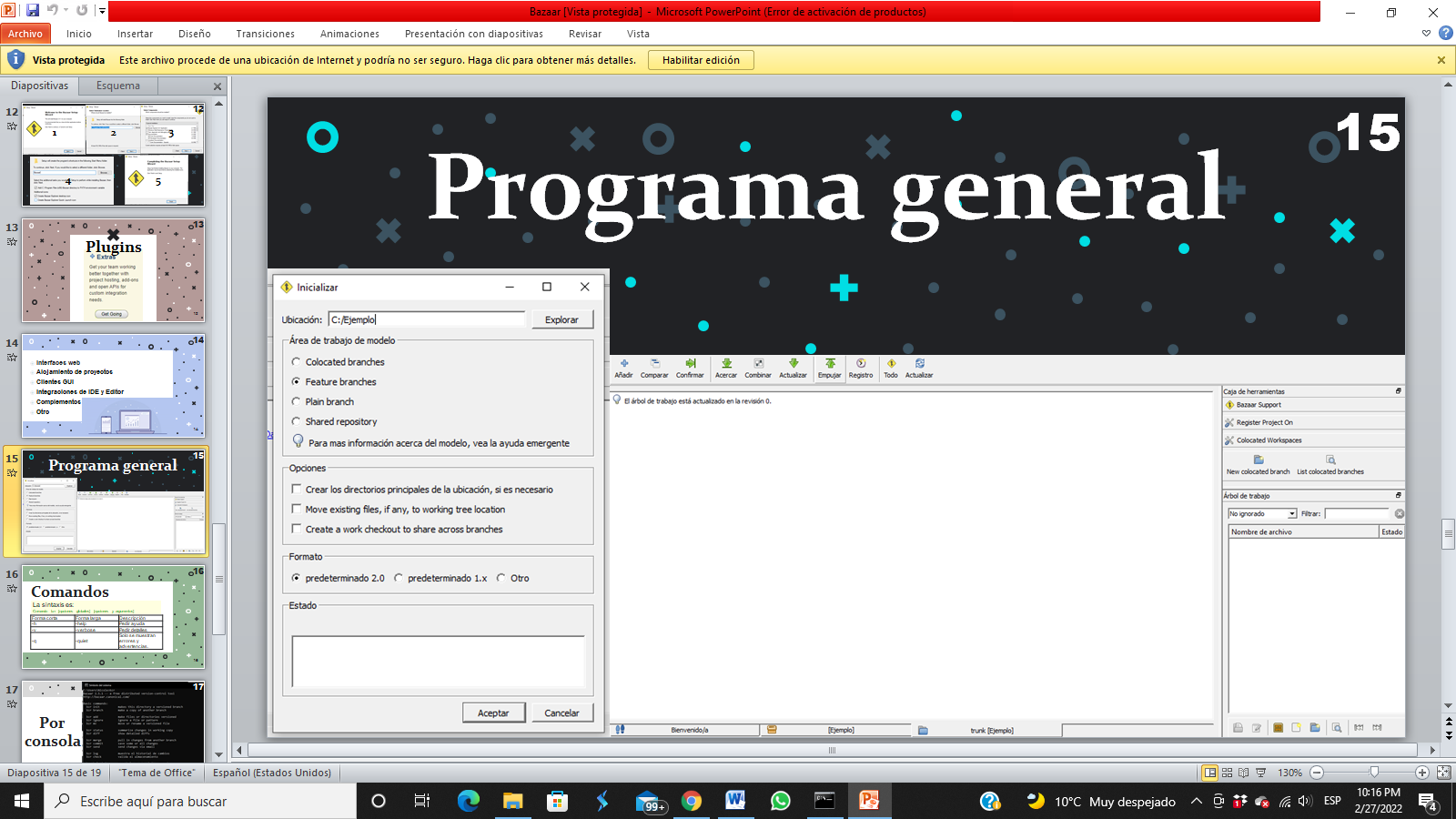
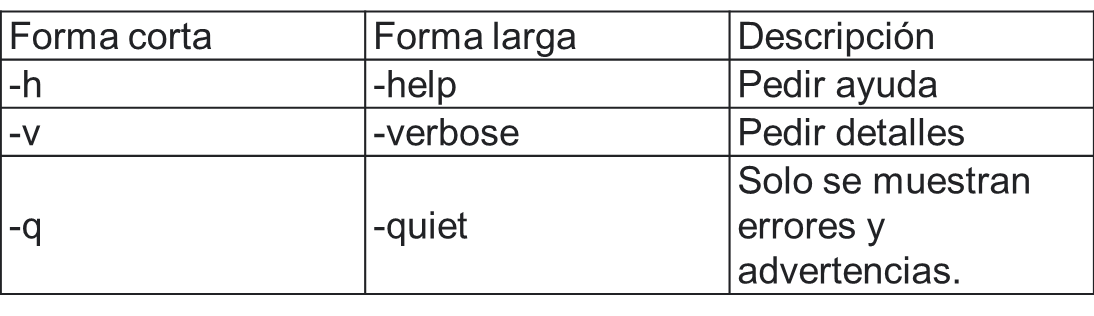
Alojamiento de proyectos

Clientes GUI

Integraciones de IDE y Editor

Complementos

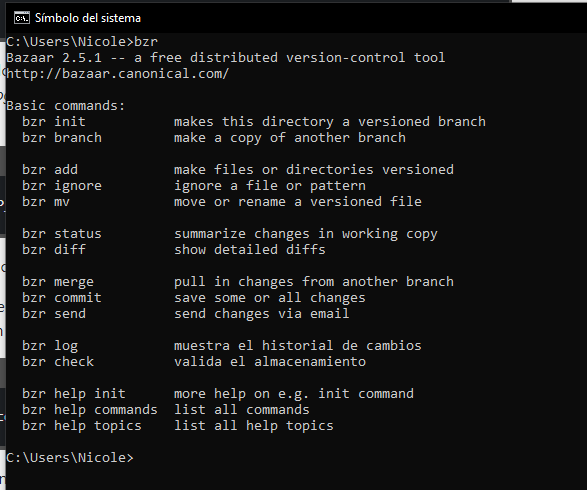
Otro



La sintaxis es:

Comando bzr [opciones globales] [opciones y argumentos]

**Comandos**



**Por**

**consola**

# Comandos

bzr whoami “Vasya Pupkin <vasya.pupkin@mail.ru>” : transfiere el nombre de usuario y el correo electrónico a Bazaar. En este caso su trabajo quedará correctamente identificado en el historial de revisiones;

bzr whoami : muestra los datos del usuario actual;

mkdir myproject : crea un proyecto llamado myproject y lo agrega al repositorio;

bzr init : proporciona a Bazaar la posibilidad de inicializar los datos en el directorio de su proyecto;

bzr add : agrega archivos al repositorio;

bzr commit – realiza un commit con una descripción (comentario). Ejemplo: bzr commit -m “Importando archivos” ;

bzr log : vea el historial de su sucursal;

bzr push : publica su sucursal;

bzr branch : crea una copia de una rama;

bzr push : publica una rama a través de ssh. Ejemplo: bzr+ssh://@bazaar.launchpad.net//miproyecto ;

bzr merge : rama que se fusiona con el lanzamiento;

bzr diff : ver los cambios después de la fusión;

bzr help: aprenda más sobre Bazaar;

bzr help commands : aprenda más comandos.

# De quien sucede Bazaar

En ocasiones se le llama Bazaar-NG (New Generation) ya que es una reimplementación de GNU Arch

El nombre le viene de un ensayo publicado por Eric S. Raimond: The Cathedral and the Bazaar: Musings on Linux and Open

Source by an Accidental Revolutionary.

En la propia web del proyecto se nos dan 10 razones para usar bazaar.

# Algunas características especiales

Hosting recomendado: https://launchpad.net/ o también http://savannah.gnu.org/

Cliente con interfaz gráfica de usuario: bazaar explorer

Unity3D es un motor de juegos en 3D que permite configurar bazaar como herramienta de control de versiones para sus proyectos

# Conclusiones

Bazaar es uno de los muchos sistemas de control de versiones que nos pueden apoyar al momento de la creación de nuestras páginas web o proyectos, después de la investigación que realizamos nos dimos cuenta que Bazaar tiene características en común con los otros SCV esto hace que el uso y el entendimiento sea más sencillo.

En conclusión Bazaar puede utilizarse como cualquier otro sistema de control de versiones pero eso ya queda a discreción de cada programador

# Referencias:

<https://bazaar.canonical.com/en/>

<https://uniwebsidad.com/libros/python/capitulo-1/instalacion-de-bazaar>

<https://es.slideshare.net/jesus.castagnetto/sistemas-para-el-control-de-versiones-de-cdigo-presentation>

<https://es.wikipedia.org/wiki/Bazaar_(software)>

<https://marvin-soto.medium.com/bazaar-6b4ca9675a68>

<https://www.dlsi.ua.es/asignaturas/dca/teoria/t10.pdf>